CHAPTER 5

VELOCITY ENERGY-EFFICIENT AND LINK-AWARE CLUSTER TREE

5.1 INTRODUCTION

This chapter deals with the scheming of a Velocity Energy-efficient and Link-aware Cluster-Tree (VELCT) for data collection in mobile wireless sensor networks which would effectively mitigate the problems of coverage distance, packet delivery, mobility, delay, traffic, tree intensity and end-to-end connection. The proposed VELCT scheme constructs the Data Collection Tree (DCT) based on the cluster head location. The Data Collection Node (DCN) in the DCT does not participate in sensing on this particular round, however, it simply collects the data packet from the cluster head and delivers it to the sink. The designed VELCT scheme minimizes energy exploitation, reduces the end-to-end delay and traffic in cluster head in wireless sensor networks by effective usage of the DCT. The strength of the VELCT algorithm is it constructs a simple tree structure, by reducing the energy consumption of the cluster head and avoids frequent cluster formation. It also maintains the cluster for a considerable amount of time.
5.2 FEATURES OF THE PROPOSED SYSTEM

To overcome the existing protocol limitations of various topologies such as energy consumption, coverage, RSS, connection time, throughput, end-to-end delay and network lifetime, we propose a novel logical topology for data collection, named, Velocity Energy-efficient and Link-aware Cluster-Tree (VELCT). It is an enhanced version of CIDT, which mitigates the existing issues in CIDT such as coverage, mobility, traffic, tree intensity and delay of the tree structure.

![VELCT Structure](image)

**Figure 5.1** VELCT Structure
Figure 5.1 shows the simple outline of our proposed scheme named into VELCT structure. It is a logical scheme, which helps to improve the performance of the network and execute effective data collection by improving the lifetime of the large scale mobile wireless sensor networks with minimum delay. VELCT is a hybrid scheme suitable for large scale dense mobile wireless sensor networks. In a mobile sensor environment, it provides superior performance than other methods.

5.3 THE VELCT DESIGN

The VELCT scheme consists of a set-up phase and steady-state phase. In the set-up phase, cluster formation and data collection tree construction is initiated to identify the optimal path between cluster member and sink. It is denoted in intra cluster and DCT communication. Later, the steady-state phase is initiated to transfer the data from the cluster members to the sink.

5.3.1 Set-up Phase

Set-up phase carries out intra cluster communication and DCT communication operations. In an intra cluster communication all the sensor node elects the cluster head with threshold value, and forms a cluster with better connection time, RSS, coverage time and robustness for connection. After the intra cluster communication, DCT communication is initiated to collect the data from its cluster head and then forwards the aggregated data packet to the sink. The cluster heads are connected with DCN, and all the DCNs connected with sink which in turn constructs the DCT.
5.3.1.1 Intra cluster communication

In ambiguous large scale wireless sensor networks, sensor nodes are densely deployed over a region. During the set-up phase, the beacon signal is used to identify the sensor nodes location and position. Once the nearby nodes are identified, cluster head election algorithm is used to elect the cluster head. The cluster head selection is based on the threshold value $U_{\varphi\alpha}^n$, connection time $\Delta D_{\varphi\alpha}^n(t, t + s)$, coverage time $\Delta I_{\varphi\alpha}^n(t, t + s)$ and robustness for connection $G_{\varphi\alpha}^n$. After the cluster head election, the next phase DCT formation is initiated.

In the proposed method, the threshold value $U_{\varphi\alpha}^n$ is calculated using equation (5.1) by adding the flag value with the multiplication of factors such as the total number of neighbor nodes, residual energy, current speed and current coverage distance of the sensor node. Let $F_C$ be the flag (set, $F_C=1$ for previous round cluster head and $F_C=0$ the sensor node having a chance to act as current round cluster head based on $U_{\varphi\alpha}^n$), $N_{\varphi\alpha}^c$ is the number of cluster members in this round, $E_{\varphi\alpha}^c$ is the current sensor node energy, $V_{\varphi\alpha}^c$ is the current speed of the sensor node, $R_{\varphi\alpha}^c$ is the current coverage radius of the sensor node, $E_{N}^m$ is the initial energy, $V_{N}^m$ is the maximum speed of the sensor node and $R_{N}^m$ is the maximum coverage radius of the sensor node. In order to avoid the election of high mobility node as cluster head, $((V_{N}^m - V_{\varphi\alpha}^c)/(V_{N}^m + V_{\varphi\alpha}^c))$ instead of $(V_{N}^m/V_{\varphi\alpha}^c)$ may be considered. Then, the $((R_{N}^m - R_{\varphi\alpha}^c)/(R_{N}^m + R_{\varphi\alpha}^c))$ is considered to elect the cluster head with maximum coverage distance. Consequently, $N_{\varphi\alpha}^c$ is the derivative from the expected number of sensor nodes $M_e$ or current number of sensor nodes $M_c$ in each cluster and that is denoted in equation (5.19). Sensor nodes having maximum number of cluster members, residual energy, RSS and connection time can be elected as
cluster head:

\[ U_{\alpha \sigma}^n = F_C + \left( \frac{N_{B0}^n}{N_{B0}^m} - \frac{N_{B0}^m}{N_{B0}^l} \times \frac{E_{N}^m - E_{G}^m}{E_{N}^l} \times \frac{V_{N}^m - V_{G}^m}{V_{N}^l + V_{G}^l} \times \frac{K_{N}^m - K_{G}^m}{K_{N}^l + K_{G}^l} \right). \]  \hspace{1cm} (5.1)

The 2-D network position of the cluster head \( \sigma \) and sensor node \( \vartheta \) at time \( t \) is characterized in equation (5.2):

\[
X_{\sigma} = x_{\sigma} + v_{\sigma} \times \cos \theta_{\sigma} \times t; \quad Y_{\sigma} = y_{\sigma} + v_{\sigma} \times \sin \theta_{\sigma} \times t, \quad Y_{\vartheta} = y_{\vartheta} + v_{\vartheta} \times \sin \theta_{\vartheta} \times t, \hspace{1cm} (5.2)
\]

where \((x,y)\) is the primary node location, \( v \) is the speed, \( \theta \) is the moving path angle between \((x,y)\), \( t \) is the connection time and \( X_{\sigma} \) or \( X_{\vartheta} \) is the current location of \( \sigma \) or \( \vartheta \) at time \( t \). Then, the subscript \((\theta, \sigma)\) corresponds to sensor node \( \vartheta \) and cluster head \( \sigma \) respectively. Let \( D_{\theta \sigma}^{\sigma}(t) \) be the distance between cluster head and cluster member at any time instance \( t \) given by

\[
[D_{\theta \sigma}^{\sigma}(t)]^2 \geq [(X_{\sigma} - X_{\vartheta})^2 + (Y_{\sigma} - Y_{\vartheta})^2]. \hspace{1cm} (5.3)
\]

At time \( t=0 \), each sensor node receives an advertisement message from any one of the cluster heads. Hence, the above 2-D network equation (5.3) is considered and simplified into

\[
[D_{\theta \sigma}^{\sigma}(t)]^2 \geq [(X_{\sigma} - X_{\vartheta})^2 + (Y_{\sigma} - Y_{\vartheta})^2], \quad \text{if } t = 0. \hspace{1cm} (5.4)
\]

Now, the connection time \( \Delta D_{\theta \sigma}^{\sigma}(t, t+s) \) is the difference between \( D_{\theta \sigma}^{\sigma}(t) \) and \( D_{\theta \sigma}^{\sigma}(t+s) \) at time instance \( t \) and \( t+s \). \( \Delta D_{\theta \sigma}^{\sigma}(t, t+s) \) can be estimated from equation (5.4):

\[
\Delta D_{\theta \sigma}^{\sigma}(t, t+s) = D_{\theta \sigma}^{\sigma}(t) - D_{\theta \sigma}^{\sigma}(t+s) \quad \text{if } (n, s) \in t, s=0,1,2, \ldots \hspace{1cm} (5.5)
\]
However, for $\Delta D_{\theta\omega}(t, t+s) = 0$, there is no sensor nodes on mobility with in a cluster. $\Delta D_{\theta\omega}(t, t+s)$ is a negative value for sensor nodes in a cluster moving away from the cluster head; $\Delta D_{\theta\omega}(t, t+s)$ is a positive value for cluster head and cluster member moving towards to each other. Now, the RSSI (Received Signal Strength Indicator) value $I$ can be calculated at any time instance $t$ and $t+s$ in equation (5.6):

$$I_{\theta\omega}(t) = I_{\theta\omega}^c(t) - I_{\theta\omega}^m$$

$$I_{\theta\omega}(t+s) = I_{\theta\omega}^c(t+s) - I_{\theta\omega}^m, \forall s \in t. \quad (5.6)$$

Where $I_{\theta\omega}^m$ is the minimum required threshold value and $I_{\theta\omega}^c$ is the current threshold value at time instance $t$ or $t+s$. If $I$ is a positive value, the cluster members join an appropriate cluster and communicate with corresponding cluster head. In this case, the coverage time $\Delta I_{\theta\omega}(t, t+s)$ is the difference between $I_{\theta\omega}(t)$ and $I_{\theta\omega}(t+s)$, which can be found using equation (5.6):

$$\Delta I_{\theta\omega}(t, t+s) = I_{\theta\omega}(t) - I_{\theta\omega}(t+s), \forall s \in t, \quad (5.7)$$

wherever, $\Delta I_{\theta\omega}(t, t+s) \geq 0$, the cluster member is moving away from the cluster head. $\Delta I_{\theta\omega}(t, t+s) \leq 0$, indicates the cluster member is moving towards the cluster head. $G_{\theta\omega}$ is the value assigned to sensor node $\theta$ for each round, which indicates its robustness for connection with cluster head $\omega$. In this case, the sensor node $\theta$ checks $G_{\theta\omega}$ with one-hop neighbor node $\omega$ to choose an optimal cluster head in every round. The dimensionless value $\delta_{\theta\omega}$, $\zeta_{\theta\omega}$, $\eta_{\theta\omega}$ and $\kappa_{\theta\omega}$ is a linear combination with constant coefficients between $0$ and $1$. The coefficients represent the significance of each factor and are denoted in equation (5.8):
\[ \delta_{\vartheta}^\alpha + \zeta_{\vartheta}^\alpha + \eta_{\vartheta}^\alpha + \kappa_{\vartheta}^\alpha = 1 \]  

(5.8)

Therefore, the above equation (5.8) can be originated into \( G_{\vartheta\alpha}^n \) and equation (5.9) represented as

\[
G_{\vartheta\alpha}^n = \left[ \left( \delta_{\vartheta}^\alpha \times \frac{E_N^m - E_c^c}{E_c^c \times N_c^c} \right) + \left( \zeta_{\vartheta}^\alpha \times \left( 1 - \frac{I_{\vartheta\vartheta}^m}{I_{\vartheta\vartheta}^c} \right) \right) \\
+ \left( \eta_{\vartheta}^\alpha \times \frac{d_{\vartheta\vartheta} - D_{\vartheta\vartheta}(t)}{D_{\vartheta\vartheta}^c(t)} \right) + \left( \kappa_{\vartheta}^\alpha \times \frac{\Delta t_{\vartheta\vartheta}^c}{t_c^f} \right) \right]
\]  

(5.9)

where \( E_N^m \) is the initial energy, \( E_c^c \) is the current energy of cluster head, \( N_c^c \) is the number of current cluster members for cluster head \( \vartheta \), \( I_{\vartheta\vartheta}^m \) is the minimum required RSSI value from \( \vartheta \) and \( \vartheta \), \( I_{\vartheta\vartheta}^c \) is the current RSSI value between \( \vartheta \) and \( \vartheta \), \( D_{\vartheta\vartheta}(t) \) is the distance between \( \vartheta \) and \( \vartheta \) at any time instance \( t \), \( d_{\vartheta\vartheta} \) is the maximum coverage distance between \( \vartheta \) and \( \vartheta \), \( \Delta t_{\vartheta\vartheta}^c \) is the estimated connection time for \( \vartheta \) begins its transmission to \( \vartheta \) and \( t_c^f \) is the current duration of the data frame for \( \vartheta \).

### 5.3.1.2 DCT communication

The DCT communication phase starts with intra cluster communication phase. In an intra cluster communication process, a sensor node elects itself as a cluster head to form a cluster. The cluster head is responsible to collect the data from its cluster members and also cluster maintenance operations (e.g., data aggregation/data fusion). Thereafter, tree formation is initiated, which connects the cluster head and sink. Now, the sink initiates the DCT formation process. Based on the location of cluster head and connection time, a few nodes are selected as DCN (Data Collection Node) to generate DCT. It is represented in DCT construction algorithm.
However, it does not participate in sensing and is not a part of any cluster on that particular round. Therefore it may act as an ordinary sensor node. In this case, the selection of DCN does not affect the data collection of a corresponding cluster. It should have better connection time with the nearest DCN and cluster head. The DCT formation is based on the location of cluster head $X_n$, connection time $\Delta D_{n2}(t, t + n)$, coverage time $\Delta I_{n2}(t, t + n)$ and robustness for connection $G_{n2}$ (i.e., between the cluster head $n$ and DCN $2$). It is elucidated in the DCN communication. After the set-up phase (i.e., behind the intra cluster and DCT communication), data transmission is initiated in the steady-state phase. Here, all the cluster members forward the data packets to sink based on the optimal path.

5.3.1.3 Data Collection Tree formation

DCT is a hierarchical tree structure, which uses DCN to collect the data from the cluster heads and deliver it to sink, thereby covering to the entire wireless sensor networks. Here, the sink selects the DCN based on the threshold value, connection time, RSS, communication range and robustness for connection. This reduces the surplus energy usage and traffic of the whole network. While the sensor nodes are on high mobility, the above selected DCN can keep the communication with the cluster head for a longer time and there is no need to update in the tree structure. In order to extend the lifetime of whole network new DCN is selected every time when the new cluster heads are elected (i.e., the new cluster head and DCN selected on every round). New DCN selection also is carried out by the sink, which is based on the mobility of the new cluster head.
1. Elect the CH over an entire network.
2. Initialize the count i, j, m=1, 7, h, ℓ=0, HC=1, NH=1, NHC=1, NH=1, NN=1.

Start

Deploy the Sensor Nodes

A

B

C

D

A. Elect a one-hop distance SN (HC=1) from the Sink as NN to generate a DCN.
B. Elect another one-hop distance SN (HC=1) as NN from previous DCN to generate a next DCN.
C. Continue the DCT construction with step A or step B to connect all the CH with DCN.

if (HC==1 && NN!=CH)

if (HC==1 && NN==CH)

Select another one-hop distance (NN, HC=1) SN from the Sink as CNI to generate a DCN.
Figure 5.2  Flow chart of DCT Construction
Figure 5.2 illustrates a flow chart which demonstrates the DCT construction procedure. The DCN collects the data from cluster head, aggregates the data (i.e., drops duplicated information) and then forwards it to the next DCN. The DCT construction algorithm is executed by sink in order to select the DCN to form an independent tree structure.

Algorithm 5.1 VELCT Algorithm

VELCT (HC, NN, NH, NHC, CNI, TIN, DCN, DCT)

set \(i, j, m=1, 7, h\)  
repeat **L1**: DCT \((m++)\)  
for \((HC=1, NHC=1, NH=1, NN=1, NN<7, h++)\)  
set \(\ell=0\)  
for \((i\leftarrow 1 \text{ to } 7\)  
if \((HC==1 \land \land NN!=CH)\) then  
set CNI=NN  
set NHC=HC++  
if \(((CNI==NN \land \land NHC==++HC \land \land NH==++NN) \land \land (NH==CH \lor NH!=CH))\) then  
set TIN[\(\ell+1\)]=CNI  
else  
return 0  
else if \((HC==1 \land \land NN==CH)\) then  
return TIN[\(\ell+1\)]=0  
end if  
set \(i=i++\)  
end for  
set best to TIN[0]
for ($j=1$ $\leq$ TIN length -1) then
  if (TIN[$j$] $>$ best) then
    set best to TIN[$j$]
  else
    set TIN to CM
  end if
  set DCN=best
end for

set $t_{DCN}^f(o)$ $\leftarrow$ DCN

if (MFD $>$ $t_{DCT}^f(\Xi)$) then
  set DCT[$h$] $\leftarrow$ DCN
  set HC=$h$+$+$
  set NN=$h$+$+$
else
  Goto L1
end if
end for
until cover all $C_H$

\textbf{Steps in DCT Construction Algorithm for VELCT:}

\begin{itemize}
  \item[(a)] Declare the variables $i, j, m, \ell$ and $h$.
  \item[(b)] Initialize the number of DCT to cover all the CHs over the network.
  \item[(c)] Assign the value to NN and iterate the value to identify the TIN from NN.
  \item[(d)] Assign the variable $\ell$ to store all CNI in TIN.
  \item[(e)] Consider 7 to identify the one-hop distance from NN.
  \item[(f)] Check the one-hop distance NN is CH or not.
  \item[(g)] Set the identity of NN as CNI.
\end{itemize}
(h) Set the NHC is one-hop distance from NN or HC.
(i) Check the CNI is NN then NHC is one-hop distance from HC, NH is the next one-hop distance from $N$ and NH is CH or not.
(j) Store CNI value into TIN array.
(k) Return null value.
(l) Check whether the value of hop count is $l$ and NN is CH.
(m) Store null value in TIN[$\ell+1$] which uses to skip the selection of CH as CNI.
(n) Increment $i$ for the next iteration to identify the next TIN.
(o) Assign TIN[0] to select the best value from TIN.
(p) Consider $j$ to decrement the array length of TIN which is used to identify the DCN.
(q) Select the best value from TIN[$j$].
(r) Assign that best value to TIN[$\ell$].
(s) Reassign the value of TIN to CM.
(t) Assign that best as DCN.
(u) Compute the frame duration of DCN as $t_{DCN}^{f}(o)$.
(v) Compare MFD and $t_{DC}^{f}(\Xi)$.
(w) Add that DCN in DCT.
(x) Now, HC is incremented from $h$, and it is considered to identify the one-hop distance SN from DCN $h$ for next DCN selection.
(y) NN is considered to one-hop distance SN from DCN $h$ for next DCN selection.
(z) Go to L1 for next DCT generation.

**Figure 5.3** Steps in DCT Construction Algorithm for VELCT
Algorithm 5.1 illustrates the DCT construction algorithm of VELCT. Initially the sink starts to find a first DCN from one-hop distance neighbor nodes to add that particular node in DCT. The parameters include HC=1 (i.e., HC is the Hop Count or Hop distance) is used to select the Current Node Identity (CNI) that is one-hop distance neighbor node (NN) from sink or DCN. Next Hop Count or Next Hop distance (NHC) is an additive value, which denotes that one-hop distance SN from the NN (i.e., NHC=++HC), and it is used to identify the cluster head from NN. The NN (Neighbor Node) is at one-hop distance SN from the sink, and Next Hop (NH) is the next one-hop distance SN from the NN.

The identified NN has been added into CNI. When the sink identifies one-hop distance neighbor node (NN) as CH, then this particular CH is skipped and DCN election is transferred to one of the neighbor nodes (i.e., HC=1, NN) from the sink or DCN as CNI. Here, the sink or DCN verifies the parameters such as threshold \( U \), connection time \( \Delta D \), coverage time \( \Delta I \) and robustness connection \( g \). After the parameter verifications, the sink selects a node with optimum value from CNI and assigns the node integrity into TIN. Now, the DCN is selected from the TIN. The DCN selection is then finalized, after which the DCN calculates the frame duration \( t_{DCN}^{f}(o) \).

After the DCN validation, the node checks the network traffic or frame duration \( t_{DCT}^{f}(\Xi) \) with Maximum Frame Duration (MFD). In the case of \( \text{MFD} > t_{DCT}^{f}(\Xi) \), the selected DCN can be utilized to generate a DCT (i.e., Step B repeated). Else, the sink skips to discover a new DCN and then starts to generate a new DCT (i.e., Step A repeated). Once the first DCN is selected, then the next DCN discovery starts from the first DCN instead of
sink. Consequently, the DCN selects another one-hop distance neighbor node to act as a CNI. Thereafter, TIN and DCN selection process is initiated to identify the next one-hop distance DCN. Likewise, DCN selection is expanded to generate the DCT to connect all of the cluster heads in the wireless sensor networks (i.e., Step C repeated).

Figure 5.4  DCT Constructions for Single Cluster

DCT construction for single cluster is shown in figure 5.4. Here, the cluster head (CH) has a chance to join with the DCN-3, DCN-4 and DCN-6. But, the CH selects a DCN with better connection time, coverage distance, robustness for connection and with less traffic (i.e., DCN-4). Likewise, the DCT also expands its tree structure with the help of DCN.
5.3.1.4 DCN communication

Once the intra cluster communication phase is completed, DCN communication phase is initiated. In DCN communication phase, sink selects the first DCN based on the threshold value $U_{pK2}$, connection time $\Delta D_{K2}(t, t + n)$, coverage time $\Delta l_{K2}(t, t + n)$, robustness for connection $g_{pK2}$ and network traffic or frame duration $t_{DCT}(Z)$ of the DCN. Thereafter, DCN starts to probe the cluster head in one-hop distance and generates a next DCN to expand the tree structure. If a cluster head is not available in one-hop distance, then it will simply elect another DCN in one-hop distance to search the cluster head and expand the tree structure. In addition, DCT makes a communication link between the cluster head and sink.

To discover the DCN in a DCT, the threshold value $U_{pK2}$ is calculated using equation (5.10). It is estimated by adding the count of prior DCN or NN from sink with multiplied factors such as the residual energy, current speed and current coverage distance of the sensor node (i.e., $\Sigma$ is assumed to identify the DCN from NN $\varphi$ and NH $\sigma$), that is

$$U_{pK2} = H_{K} + \left( \frac{E_{N}^{m} - E_{N}^{c}}{E_{N}^{m}} \times \frac{V_{N}^{m} - V_{c}^{c}}{V_{N}^{m} + V_{c}^{c}} \times \frac{R_{N}^{m} - R_{N}^{c}}{R_{N}^{m} + R_{N}^{c}} \right),$$

(5.10)

where $H_{K}$ is the hop count of $K$ from sink in DCT (i.e., $K$ is supposed to notice the prior DCN or NN in DCT), $E_{N}^{c}$ is the current energy of $\Sigma$, $V_{c}^{c}$ is the current speed of $\Sigma$, $E_{N}^{m}$ is the initial energy of sensor node, $V_{N}^{m}$ is the maximum speed of the sensor node, $R_{N}^{c}$ is the current coverage radius of $\Sigma$ and $R_{N}^{m}$ is the maximum coverage radius of the sensor node.
Let us consider that the 2-D network position of the sink $\lambda$, NN $\varphi$, NH or CH $\sigma$ and DCN $\varrho$ at time $t$ is

$$X_\varphi = x_\varphi + v_\varphi \cos \theta_\varphi t; \quad Y_\varphi = y_\varphi + v_\varphi \sin \theta_\varphi t$$

$$X_\sigma = x_\sigma + v_\sigma \cos \theta_\sigma t; \quad Y_\sigma = y_\sigma + v_\sigma \sin \theta_\sigma t$$

(5.11)

$$X_\varrho = x_\varrho + v_\varrho \cos \theta_\varrho t; \quad Y_\varrho = y_\varrho + v_\varrho \sin \theta_\varrho t.$$

The sink position $(X_\lambda, Y_\lambda)$ is considered to be fixed, and located outside of boundary of the sensing region. On each round, the movement position of NN $\varphi$, NH $\sigma$ and DCN $\varrho$ is calculated using equation (5.11). Let $D_{\varrho \varphi}^\varphi(t)$ be the distance between sink $\lambda$ and NN $\varphi$, $D_{\varphi \varphi}^\varphi(t)$ is the distance between DCN $\varrho$ and NN $\varphi$ and $D_{\varphi \varphi}^\sigma(t)$ is the distance between NN $\varphi$ and NH $\sigma$, and can be denoted at any time instance $t$ in

$$[D_{\varrho \varphi}^\varphi(t)]^2 \geq [(X_\lambda - X_\varphi)^2 + (Y_\lambda - Y_\varphi)^2]$$

$$[D_{\varphi \varphi}^\varphi(t)]^2 \geq [(X_\varrho - X_\varphi)^2 + (Y_\varrho - Y_\varphi)^2], \quad 0 \leq t \leq n.$$  

(5.12)

At time $t=0$, the distance $D_{\varrho \varphi}^\varphi(t)$, $D_{\varphi \varphi}^\varphi(t)$ and $D_{\varphi \varphi}^\sigma(t)$ in equation (5.12) can be considered as
\[ [D_{\phi\phi}^g(t)]^2 \geq [(x_\lambda - x_\phi)^2 + (y_\lambda - y_\phi)^2] \]

\[ [D_{\phi\phi}^g(t)]^2 \geq [(x_\psi - x_\phi)^2 + (y_\psi - y_\phi)^2] \]  \hspace{1cm} (5.13)

\[ [D_{\phi\phi}^g(t)]^2 \geq [(x_\phi - x_\mu)^2 + (y_\phi - y_\mu)^2], \hspace{0.5cm} t=0, \forall n \in t. \]

Consequently, \( \aleph \) has been considered instead of \( \lambda, \rho \) and \( \phi \) (i.e., it represents that sink or DCN or NN), \( \Sigma \) as a substitute of \( \varphi \) and \( \sigma \) (i.e., it signifies that NN or NH). Let \( \Delta D_{\aleph\Sigma}^2(t, t+n) \) be the diversity among \( D_{\aleph\Sigma}^2(t) \) and \( D_{\aleph\Sigma}^2(t+n) \), and is represented for any time instance between \( t \) and \( t+n \) in

\[ \Delta D_{\aleph\Sigma}^2(t, t+n) = D_{\aleph\Sigma}^2(t) - D_{\aleph\Sigma}^2(t+n), \hspace{0.5cm} t=0,1,2,...,n, \forall n \in t. \]  \hspace{1cm} (5.14)

However, \( \Delta D_{\aleph\Sigma}^2(t, t+n) = 0 \), signifies both nodes (i.e., \( \aleph \) and \( \Sigma \)) are not in mobility and is separated in even distance. \( \Delta D_{\aleph\Sigma}^2(t, t+n) \) is the negative value for both the nodes moving away and \( \Delta D_{\aleph\Sigma}^2(t, t+n) \) is the positive value for both nodes moving towards each other.

Figure 5.5 shows the distances estimation DCN to construct DCT. The Received Signal Strength Indicator (RSSI) value \( \iota \) between any two nodes, at the time instance \( t \) and \( t+n \) is calculated in equation (5.15):

\[ I_{\aleph\Sigma}^2(t) = I_{\aleph\Sigma}^c(t) - I_{\aleph\Sigma}^m \]

\[ I_{\aleph\Sigma}^2(t+n) = I_{\aleph\Sigma}^c(t+n) - I_{\aleph\Sigma}^m, \hspace{0.5cm} \forall n \in t. \]  \hspace{1cm} (5.15)
Figure 5.5 Distances estimation in DCN to Construct DCT

(a), (b) Selection of first DCN from the sink. (c), (d) Selection of second DCN from first DCN on DCT. (e) Selection of \((O-1)^{th}\) DCN on DCT.
Here, $I_{N2}^m$ is the minimum required threshold value and $I_{N2}^c$ is the current threshold value at time instance $t$ or $t + n$. If $I_{N2}^m$ is a positive value, then the $N$ has a likelihood to join with nearest $Z$, and this is used to establish a communication with corresponding nodes. $\Delta I_{N2}^2(t, t + n)$ can be found using equation (5.15) as follows.

$$\Delta I_{N2}^2(t, t + n) = I_{N2}^3(t) - I_{N2}^3(t + n), \quad \forall n \in t. \quad (5.16)$$

Wherever, $\Delta I_{N2}^2(t, t + n) \geq 0$, then both nodes move away from their current position. If $\Delta I_{N2}^2(t, t + n) \leq 0$, then both nodes move towards each other. The dimensionless value $\delta_{N2}^2$, $\zeta_{N2}^2$, $\eta_{N2}^2$ and $\kappa_{N2}^2$ is a linear combination with constant coefficients between $0$ and $1$. The coefficients represent the significance of each factor and denoted as,

$$\delta_{N2}^2 + \zeta_{N2}^2 + \eta_{N2}^2 + \kappa_{N2}^2 = 1. \quad (5.17)$$

$G_{N2}^p$ is the value assigned to all $N$ in each round, which indicates its robustness for connection with $Z$:

$$G_{N2}^p = \left[ \left( \delta_{N2}^2 \times \frac{E_{N}^m - E_{Z}^c}{E_{Z}^c} \right) + \left( \zeta_{N2}^2 \times \left( 1 - \frac{I_{N2}^m}{I_{N2}^c} \right) \right) \right. \left. + \left( \eta_{N2}^2 \times \frac{d_{N2}^2 - D_{N2}^2(t)}{D_{N2}^2(t)} \right) + \left( \kappa_{N2}^2 \times \frac{\Delta t_{N2}^2}{t_{DCT}(\Xi)} \right) \right], \quad (5.18)$$

where $E_{N}^m$ is the initial energy of $N$, $E_{Z}^c$ is the current energy of $Z$, $I_{N2}^m$ is the minimum required RSSI to make a connection from $N$ and $Z$, $I_{N2}^c$ is the current RSSI to establish a connection between $N$ and $Z$, $d_{N2}^2$ is the maximum coverage distance between $N$ and $Z$, $D_{N2}^2(t)$ is the distance between $N$ and $Z$ at
any time instance $t,$ $\Delta t^2_{\text{RC}}$ is the estimated connection time for $\mathfrak{N}$ beginning its transmission to $\mathfrak{Z}$ and $t^f_{\text{DCT}}(\Xi)$ is the $\mathfrak{Z}^{\text{th}}$ current duration of a data frame.

### 5.3.1.5 Frame duration

In VELCT, the number of current cluster members $M_c$ and the number of expected cluster members $M_e$ in each round can be obtained using equation 5.19.

$$M_c = M_e - (M_{nj} + M_d + M_s)$$

$$M_e = \frac{N_c - C_H - C_T}{C_H}; \quad N_c = N_t - N_d,$$

where $M_c$ is the current cluster member from one cluster, $M_e$ is the expected number of cluster members, $M_{nj}$ is the newly joined cluster member from neighbor cluster in this round, $M_d$ is the number of cluster members that are considered dead, $M_s$ is the total number of cluster member on sleep state, $N_c$ is the total number of current sensor nodes, $N_t$ is the total number of sensor nodes over a network, $N_d$ is the number of sensor nodes considered dead, $C_H$ is the cluster head and $C_T$ is the total number of DCN. Therefore, the current duration of the data frame $t^f_c$ from each cluster is estimated using equation (5.20):

$$t^f_c = \frac{L_p}{R_b} \times M_c \quad \text{or} \quad t^f_c = \frac{L_p}{R_b} \times M_e,$$

where $L_p$ is the data packet length and $R_b$ is the transmission bit rate.
Let the current data frame duration of any $o^{th}$ DCN $t_{DCN}^f(o)$ be represented by equation (5.23). $t_{DCN}^f(Y)$ is the data frame duration of DCN to DCN and $t_{DCN}^f(\Psi)$ is the cluster head to DCN data frame duration in equation (5.21):

$$t_{DCN}^f(Y) = \sum_{\nu=1}^{\nu} \frac{L_C^\nu(y)}{R_C^\nu(y)} \times C_T(y), \quad y \in \Psi$$

$$t_{DCN}^f(\Psi) = \sum_{\psi=1}^{\psi} \frac{L_C^\psi(\psi)}{R_C^\psi(\psi)} \times C_H(\psi), \quad \psi \in \Psi,$$  \hspace{1cm} (5.21)

where $L_C^\nu(y)$ is the $\nu^{th}$ DCN packet length, $L_C^\psi(\psi)$ is the packet length of $\psi^{th}$ cluster head, $R_C^\nu(y)$ is the $\nu^{th}$ DCN transmission bit rate, $R_C^\psi(\psi)$ is the transmission bit rate for $\psi^{th}$ cluster head, $C_T(y)$ is the number of DCN and $C_H(\psi)$ is the number of cluster head. Likewise, $t_{DCN}^f(\Phi)$ is the sum of frame duration of the connected one-hop distance DCN $t_{DCN}^f(\Pi)$ and cluster head $t_{DCN}^f(\Delta)$, and that is represented in equation (5.22):

$$t_{DCN}^f(\Phi) = t_{DCN}^f(\Pi) + t_{DCN}^f(\Delta), \quad \forall(\Pi, \Delta) \in \Phi,$$

$$t_{DCN}^f(\Phi) = \left\{ \sum_{\pi=1}^{\pi} \left( \frac{L_C^\pi(\pi)}{R_C^\pi(\pi)} \times C_T(\pi) \right) \right\} + \left\{ \sum_{\delta=1}^{\delta} \frac{L_C^\delta(\delta)}{R_C^\delta(\delta)} \times C_H(\delta) \right\}.$$  \hspace{1cm} (5.22)

Where $L_C^\pi(\pi)$ is the $\pi^{th}$ DCN packet length, $R_C^\pi(\pi)$ is the $\pi^{th}$ DCN transmission bit rate, $C_T(\pi)$ is the number of DCN, $L_C^\delta(\delta)$ is the packet length of $\delta^{th}$ cluster head, $R_C^\delta(\delta)$ is the transmission bit rate for $\delta^{th}$ cluster head, $C_H(\delta)$ is the number of cluster heads. Now, each DCN may calculate the frame duration $t_{DCN}^f(o)$ in equation (5.23). In this case, the DCN may
have the frame duration of either \( t_{DCN}^{f}(\Upsilon) \) or \( t_{DCN}^{f}(\Psi) \) or \( t_{DCN}^{f}(\Phi) \) (i.e., the DCN may connected with the one-hop distance DCN or cluster head or DCN and cluster head). The total frame duration of any DCN in DCT \( t_{DCT}^{f}(0) \) is computed in equation (5.24):

\[
t_{DCN}^{f}(o) = t_{DCN}^{f}(\Upsilon) \text{ or } t_{DCN}^{f}(\Psi) \text{ or } t_{DCN}^{f}(\Phi), \quad \forall (\Upsilon, \Psi, \Phi) \in o
\]

\[
t_{DCT}^{f}(0) = \sum_{o=1}^{0} t_{DCN}^{f}(o), \quad o \in 0.
\]

Similarly, the frame duration of \( t_{DCT}^{f}(P) \) is an additive value which is used to discover the frame duration of any DCT \( t_{DCT}^{f}(\Xi) \). Therefore, we consider that \( P = 0 \) in the equation (5.24) and that is denoted as follows:

\[
t_{DCT}^{f}(P) = \sum_{\zeta=1}^{P} t_{DCN}^{f}(\zeta), \quad \zeta \in P.
\]

The DCT frame duration \( t_{DCT}^{f}(\Xi) \) (i.e., frame duration of the first DCN from the sink on DCT) computed from equation (5.24) and equation (5.25) is used in equation (5.26). The DCN frame duration estimation starts from the bottom of DCN of the DCT to the top of sink.

\[
t_{DCT}^{f}(\Xi) = t_{DCT}^{f}(0) - t_{DCT}^{f}(P)
\]

\[
t_{DCT}^{f}(\Xi) = \sum_{o=0}^{1} t_{DCN}^{f}(o) - \sum_{\zeta=P-1}^{1} t_{DCN}^{f}(\zeta), \quad \forall (o, \zeta) \in \Xi.
\]

Consequently, \( t_{DCT}^{f}(\Xi) \) is further simplified as illustrated in equation (5.27)
\[ t_{DCT}^f(\Xi) = \sum_{o=0}^{1} \sum_{c=p-1}^{1} [ t_{DCN}^f(o) - t_{DCN}^f(c) ]. \]  

(5.27)

Now, the sink checks the \( t_{DCT}^f(\Xi) \) with Maximum Frame Duration (MFD) to avoid the surplus traffic in each DCN of DCT. Then the hierarchical order of DCN frame duration of DCT is given by equation (5.28)

\[ \text{MFD} > t_{DCT}^f(\Xi) \approx t_{DCT}^f(0) > t_{DCN}^f(o). \]  

(5.28)

Figure 5.6 Frame duration hierarchy of DCT
Figure 5.6 presents the frame duration hierarchy of DCT. Let us consider a DCN in DCT, that is able to communicate with the one-hop either DCN (i.e., π or ν) or CH (i.e., ψ or δ) sends its data to a next level DCN (i.e., Φ or Ψ or Υ) and that aggregates the data from all its neighbor, and then assign the integrity of the particular DCN as o before forwarding them. Likewise, each DCN can perform the above similar operations and compute the total frame duration $t^{f}_{DCN}(o)$ of that particular round. Consequently, the sink estimates the total frame duration $t^{f}_{DCT}(Ξ)$ of any DCT and compares with the MFD. Let us consider the above equation (5.28), if $MFD > t^{f}_{DCT}(Ξ)$, the sink selects a new DCN in the same DCT, and in case of $MFD \leq t^{f}_{DCT}(Ξ)$ sink terminates the DCN selection process and it initiates a new DCT in order to avoid the packet dropping over the DCT.

5.3.2 Steady-State Phase

Once the set-up phase is completed, steady-state phase is initiated. In steady-state phase, all the cluster members send the collected data to the cluster head in their allocated time slots. Later, the cluster head starts to collect and aggregate the data from its cluster members. Meanwhile, the DCT communication is also initiated, and it uses direct sequence spread spectrum to transfer the data from the cluster head to DCN and then the sink. Here, the DCN collects and aggregates the data from the corresponding cluster head or DCN.

5.4 PERFORMANCE EVALUATION

In this section, performance of the existing algorithms under various parameter settings has been presented through simulation. Network
Simulator 2 (NS-2) is used to carry out the performance study of VELCT with respect to some well-evaluated existing protocols namely, CIDT, MBC, CTDGA, CREEC and EEDCP-TB.

5.4.1 Scenario Setup

Wireless sensor system comprising of 500 nodes is used in the simulation scenario. All the nodes are randomly deployed in a square region of 1000×1000 m². The size of the data packet is 256 bytes, the transmission range within the cluster is set as 40m, the sensing range is 20m, and the base station is located in (x=500, y=1050). Further, communication energy parameters is set as $E_{elec} = 50$ nJ/bit/m² and $E_{amp} = 0.0013$ pJ/bit/m⁴ (Heinzelman et al 2002). The energy required for data aggregation is set as $E_{DA} = 50$ nJ/bit/signal. The sensing range is 20 m. The transmission range among the sensor nodes within the cluster is 40m. The random waypoint model has been selected for assigning the sensor node mobility. The simulation parameters have been listed in the table 5.1.

Table 5.1  Network simulation parameters

<table>
<thead>
<tr>
<th>Parameters</th>
<th>Values</th>
</tr>
</thead>
<tbody>
<tr>
<td>Area</td>
<td>1000 × 1000 m²</td>
</tr>
<tr>
<td>Base station location</td>
<td>(x=500, y=1050)</td>
</tr>
<tr>
<td>Number of nodes</td>
<td>500</td>
</tr>
<tr>
<td>$E_{elec}$</td>
<td>50 nJ/bit/m²</td>
</tr>
<tr>
<td>$E_{amp}$</td>
<td>0.0013 pJ/bit/m⁴</td>
</tr>
</tbody>
</table>
### Parameters and Values

<table>
<thead>
<tr>
<th>Parameters</th>
<th>Values</th>
</tr>
</thead>
<tbody>
<tr>
<td>$E_{DA}$</td>
<td>50 nJ/bit/signal</td>
</tr>
<tr>
<td>Data packet size</td>
<td>256 bytes</td>
</tr>
<tr>
<td>Transmission range within cluster</td>
<td>40 m</td>
</tr>
<tr>
<td>Sensing range</td>
<td>20 m</td>
</tr>
<tr>
<td>Mobility</td>
<td>5 m/s to 30 m/s</td>
</tr>
<tr>
<td>Mobility Model</td>
<td>Random Waypoint Model</td>
</tr>
<tr>
<td>Simulation time</td>
<td>100 s</td>
</tr>
</tbody>
</table>

### 5.4.2 Influence on number of nodes

The variations in number of mobile sensor nodes results in a great impact on the performance of large scale mobile sensor networks. The mobile sensor node density is varied from the initial setup of 100 to 500 mobile sensor nodes. Consequently, the maximum speed of mobile sensor nodes could be varied up to 20 m/s. As this gets varied, the performance of each routing scheme in terms of packet delivery ratio, throughput, total energy and delay has been evaluated. The results are illustrated in figures 5.7 to 5.10.

#### 5.4.2.1 Packet Delivery Ratio with respect to Number of nodes

Figure 5.7 illustrates the performance of VELCT algorithm with CIDT, MBC, CTDGA, CREEC and EEDCP-TB in terms of PDR. It may be
noted from figure 5.7 that VELCT achieves 2% (approx) better performance than CIDT in terms of PDR. It may also be noted that VELCT achieves 5% (approx), 6% (approx), 8% (approx) and 10% (approx) better performance than MBC, CREEC, CTDGA and EEDCP-TB respectively in terms of PDR. The major reason why VELCT achieves better performance than other algorithms in terms of PDR is because, VELCT provides considerably stable links than other existing algorithms. This is because of VELCT reducing the packet overhead of the cluster head in each round, and selects stable links with maximum connection time and RSS. Furthermore, DCT offers less traffic (i.e., minimum load over the network) between the cluster head and the sink. An analysis on the various protocols has been presented in table 5.2.

![Figure 5.7 Packet Delivery Ratio with respect to Number of nodes](image-url)
Table 5.2  Packet Delivery Rate with respect to Number of nodes

<table>
<thead>
<tr>
<th>Number of nodes</th>
<th>Packet Delivery Rate (in Percentage)</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>EEDCP-TB</td>
</tr>
<tr>
<td>100</td>
<td>95.00</td>
</tr>
<tr>
<td>200</td>
<td>92.50</td>
</tr>
<tr>
<td>300</td>
<td>89.50</td>
</tr>
<tr>
<td>400</td>
<td>88.00</td>
</tr>
<tr>
<td>500</td>
<td>86.50</td>
</tr>
</tbody>
</table>

5.4.2.2  Throughput with respect to Number of nodes

Observations from figure 5.8 clearly indicate that VELCT algorithm achieves better throughput when compared to the existing schemes. This may be attributed to two salient features of the VELCT algorithm: First, VELCT offers minimum load on intra cluster and inter cluster communications and Second, VELCT avoids unwanted control packet flooding on node mobility because DCT selects an optimal link between the cluster head and sink. The total number of nodes in each cluster is also properly maintained for every rounds. The comparative ranges of the proposed protocols have been depicted in table 5.3.
Figure 5.8 Throughput with respect to Number of nodes

Table 5.3 Throughput with respect to Number of nodes

<table>
<thead>
<tr>
<th>Number of nodes</th>
<th>Throughput (in Mbps)</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>EEDCP-TB</td>
</tr>
<tr>
<td>100</td>
<td>0.830</td>
</tr>
<tr>
<td>200</td>
<td>0.800</td>
</tr>
<tr>
<td>300</td>
<td>0.773</td>
</tr>
<tr>
<td>400</td>
<td>0.741</td>
</tr>
<tr>
<td>500</td>
<td>0.700</td>
</tr>
</tbody>
</table>
5.4.2.3  Total Energy with respect to Number of nodes

Figure 5.9 illustrates the performance of every protocols in terms of total number of nodes and energy consumption. VELCT selects cluster head with better threshold value, connection time, RSSI and minimum control packets overhead. Each cluster head is selected with maximum residual energy, coverage distance and less mobility. In terms of total energy consumption, VELCT scheme saves upto 31% (approx), 36% (approx), 41% (approx), 45% (approx) and 65% (approx) energy with respect to CIDT, MBC, CTDGA, CREEC and EEDCP-TB respectively. This is because of VELCT selecting the cluster head with better threshold value by reducing the convention time and control packet overhead, and RSSI is also maintained. VELCT elects the cluster head with maximum residual energy, better coverage distance and lowest mobility thereby reducing total energy consumption for the entire wireless sensor network. The simulated values corresponding to each protocol has been provided in table 5.4.

Figure 5.9  Total Energy with respect to Number of nodes
Table 5.4  Total Energy with respect to Number of nodes

<table>
<thead>
<tr>
<th>Number of nodes</th>
<th>Total Energy (in mJ)</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>EEDCP-TB</td>
</tr>
<tr>
<td>100</td>
<td>250</td>
</tr>
<tr>
<td>200</td>
<td>272</td>
</tr>
<tr>
<td>300</td>
<td>299</td>
</tr>
<tr>
<td>400</td>
<td>399</td>
</tr>
<tr>
<td>500</td>
<td>410</td>
</tr>
</tbody>
</table>

5.4.2.4  Delay with respect to Number of nodes

Figure 5.10 shows the performance of all the protocols in terms of delay with respect to total number of sensor nodes. 100 to 500 sensor nodes have been selected to investigate the performance of delay in the network. The simulation results show that, VELCT depicts minimum delay when compared to other existing protocols such as CIDT, MBC, CTDGA, CREEC and EEDCP-TB. This is because, VELCT offers shortest path. It also provides a stable link with maximum connection time between cluster members to the sink for each round, which reduces the packet drop ratio and packet retransmissions over the network. The comparative range of each protocol has been shown in table 5.5.
Figure 5.10  Delay with respect to Number of nodes

Table 5.5  Delay with respect to Number of nodes

<table>
<thead>
<tr>
<th>Number of nodes</th>
<th>EEDCP-TB</th>
<th>CTDGA</th>
<th>CREEC</th>
<th>MBC</th>
<th>CIDT</th>
<th>VELCT</th>
</tr>
</thead>
<tbody>
<tr>
<td>100</td>
<td>6.00</td>
<td>4.20</td>
<td>5.10</td>
<td>3.90</td>
<td>2.90</td>
<td>2.50</td>
</tr>
<tr>
<td>200</td>
<td>7.00</td>
<td>5.30</td>
<td>6.20</td>
<td>4.80</td>
<td>3.80</td>
<td>2.65</td>
</tr>
<tr>
<td>300</td>
<td>8.10</td>
<td>6.50</td>
<td>7.30</td>
<td>6.20</td>
<td>4.60</td>
<td>2.75</td>
</tr>
<tr>
<td>400</td>
<td>9.20</td>
<td>7.80</td>
<td>8.70</td>
<td>7.10</td>
<td>5.40</td>
<td>2.90</td>
</tr>
<tr>
<td>500</td>
<td>10.00</td>
<td>9.00</td>
<td>9.70</td>
<td>8.30</td>
<td>6.50</td>
<td>3.00</td>
</tr>
</tbody>
</table>
5.4.3 Influence on sensor nodes mobility

To investigates the performance of the proposal algorithm in terms of mobility, two parameters have been measured namely PDR and delay with respect to speed. The speed of the mobile sensor node is varied from 5 m/s to 30 m/s for each protocols. Figures 5.11 and 5.12 clearly show that the performance evaluation of VELCET interns of packet delivery ratio and delay with respect to speed. These parameters measured and compared for evaluating the performance of VELCET, CIDT, MBC, CTDGA, CREEC and EEDCP-TB protocols.

5.4.3.1 Packet Delivery Ratio with Speed

![Packet Delivery Ratio with respect to Speed](image)

Figure 5.11 illustrated that VELCET achieves superior performance when compared to CIDT, MBC, CTDGA, CREEC and EEDCP-
TB in terms of packet delivery ratio with respect to speed. From the simulation results, it could be clearly understand that VELCT protocol provides a stable link and adopts itself to high mobility. It is observed that the proposed VELCT protocol conserves the sensor node residual energy, prolongs the network lifetime and thereby increases network reliability. Additionally, adapts to highly mobile environments and provides better quality for communication. The values corresponding to every protocol have been depicted in table 5.6.

Table 5.6  Packet Delivery Ratio with respect to Speed

<table>
<thead>
<tr>
<th>Speed (in m/s)</th>
<th>EEDCP-TB</th>
<th>CTDGA</th>
<th>CREEC</th>
<th>MBC</th>
<th>CIDT</th>
<th>VELCT</th>
</tr>
</thead>
<tbody>
<tr>
<td>05</td>
<td>94.00</td>
<td>96.00</td>
<td>96.70</td>
<td>97.14</td>
<td>98.27</td>
<td>99.54</td>
</tr>
<tr>
<td>10</td>
<td>91.00</td>
<td>93.00</td>
<td>94.13</td>
<td>95.33</td>
<td>96.77</td>
<td>99.00</td>
</tr>
<tr>
<td>15</td>
<td>89.50</td>
<td>91.00</td>
<td>92.23</td>
<td>93.34</td>
<td>95.75</td>
<td>98.80</td>
</tr>
<tr>
<td>20</td>
<td>87.00</td>
<td>88.30</td>
<td>89.30</td>
<td>92.00</td>
<td>94.92</td>
<td>98.50</td>
</tr>
<tr>
<td>25</td>
<td>85.50</td>
<td>87.00</td>
<td>88.37</td>
<td>90.47</td>
<td>93.55</td>
<td>98.30</td>
</tr>
<tr>
<td>30</td>
<td>83.00</td>
<td>85.50</td>
<td>87.12</td>
<td>89.04</td>
<td>92.50</td>
<td>98.10</td>
</tr>
</tbody>
</table>

5.4.3.2  Delay with respect to Speed

Figure 5.12 illustrates the performance of VELCT with EEDCP-TB, CREEC, CTDGA, MBC and CIDT in terms of PDR. It is seen that,
VELCT achieves better stability than the existing protocols in highly mobile sensor node scenarios. The simulated values have been tabulated in table 5.7. The end-to-end delay slightly increases with the increase in the speed of sensor nodes for all the six protocols. However, in VELCT it is worth to be noted that the delay is more gradual than other protocols as illustrated in figure 5.12.

![Table 5.7 Delay with respect to Speed](image)

<table>
<thead>
<tr>
<th>Speed (in m/s)</th>
<th>EEDCP-TB</th>
<th>CTDGA</th>
<th>CREEC</th>
<th>MBC</th>
<th>CIDT</th>
<th>VELCT</th>
</tr>
</thead>
<tbody>
<tr>
<td>05</td>
<td>3.60</td>
<td>5.44</td>
<td>6.14</td>
<td>4.00</td>
<td>3.03</td>
<td>1.50</td>
</tr>
<tr>
<td>10</td>
<td>9.00</td>
<td>7.18</td>
<td>8.16</td>
<td>6.12</td>
<td>4.13</td>
<td>2.10</td>
</tr>
<tr>
<td>15</td>
<td>13.12</td>
<td>9.02</td>
<td>9.88</td>
<td>7.36</td>
<td>5.00</td>
<td>2.60</td>
</tr>
<tr>
<td>20</td>
<td>19.06</td>
<td>10.27</td>
<td>11.00</td>
<td>8.60</td>
<td>7.00</td>
<td>3.00</td>
</tr>
<tr>
<td>25</td>
<td>16.36</td>
<td>13.31</td>
<td>15.51</td>
<td>11.57</td>
<td>8.45</td>
<td>3.40</td>
</tr>
<tr>
<td>30</td>
<td>22.00</td>
<td>17.37</td>
<td>19.06</td>
<td>13.12</td>
<td>9.00</td>
<td>3.60</td>
</tr>
</tbody>
</table>

5.5 CHAPTER SUMMARY

As the impact of wireless sensor network on real-time civil and military applications increases, more number of sensor nodes are required to monitor large scale areas. In this chapter, an efficient method VELCET has
been proposed to construct a mobility-based network management architecture for wireless sensor networks by exploiting the network lifetime, connection time, residual energy, RSSI, throughput, PDR and stable link for mobile sensor nodes. Every cluster member chooses the cluster head with better connection time and forwards the data packets to the corresponding cluster head in its allocated time slot. Similarly, the sink or DCN elects the one-hop neighbor DCN or cluster head with maximum threshold value, connection time, RSSI and reduced network traffic. From the simulation results, it could be clearly seen that VELCT offers more stable links, better throughput, reduce energy utilization and PDR with reduced network traffic, than existing protocols such as EEDCP-TB, CREEC, CTDGA, MBC and CIDT.